CHAPTER 4

PROPOSED APPROACHES

We are proposing different approaches to avoid the problems of Virtual Machines in the cloud computing environs.

4.1 Security mechanism between VMM and VMs (1st Approach):

The first approach is to put on a safety mechanism within the virtualized computing environment, to screen the stream of traffic flow between Virtual Machine Monitor and one or more virtual machines. The upper part of the diagram is the one where we don’t have any security system. To make the virtual environment safe, the lower part is provided with the security system which is called as Intrusion Detection System and Intrusion Prevention System. It is shown in figure given below:

![Figure 3: Security mechanism applied between VMs and VMM](image-url)
The security mechanism ie., Intrusion Detection and Prevention System (IDS/IPS) system provides protection against attacks originating on the network. This protection mechanism checks each and every request and reply message that is communicated between Virtual Machines and the Virtual Machine Monitor. When they receive each and every message, they check for malwares, threats and any other type of problems which may present in the messages. If it encounters any such problem, then they immediately avoids that infected message to transfer in between Virtual machine and Virtual machine monitor to avoid the problems which may arise with the virtual machine environment. The system try to make those messages free from infections. But there are some significant limitations to this option:

- Security mechanism is available between Virtual Machine and Virtual Machine Monitor, which cannot prevent attacks, threats or malwares between Virtual Machines.

- If we enable the Virtual Machines to move from one physical machine to other, the security setting is lost. It is important to arrange the gathering of security apparatuses for each potential goal to which a Virtual Machine could be moved, bringing about a comparing negative effect on execution.

- The Security mechanism must handle all the communication traffic between all the Virtual Machines, Virtual Machine Monitor and the network, this will result in a performance degradation.

- There are chances of malwares, threats and viruses which may injected from outside directly to the virtual machines or to the Virtual Machine Monitor, which makes virtual machines and virtual machine monitor to behave abnormally by consuming more computing resources such as processing power, memory, network etc., This cannot be identified by the security mechanism which is installed between the different virtual machines and the virtual machine monitor.

These are the limitations which makes the security mechanism that is installed between virtual machines and virtual machine monitor is not in the position to provide the security to whole virtual environment, to make virtual machines to work normally without consuming more computing resources.
This approach is simulated using Cloudsim simulator tool. The results are given in Result section. This is the simplest approach, which provides security for the transactions between different virtual machines and Virtual machine monitor. Here virtual machines are created using cloudsim and allow them to interact with VMM for resources allocation. We make them to send the requests to VMM for more resources such as processor time, memory etc., than they actually required. This cannot be handled by IDS/IPS system. Since it is there for avoiding the threats, it can detect only the transactions that are infected. It cannot identify the requests from VMs for more resources than actually required.
4.2 Centralized Security Supervisor between VMM and VMs (2nd Approach):

The second approach is to present a novel means of security control, which is shown in the figure given below i.e., Security Supervisor between Virtual Machine Monitor and Virtual Machines. Security Supervisor functions utilize self-assessing APIs to access privileged state information about each Virtual Machine, such as the consumption of processing power, their memory utilization, the way in which they send and receive the request-reply messages, state and network traffic etc. This removes the problems of previous approach such as, Inter-Virtual Machine and non-transparency restrictions of the Security-mechanism for IDS/IPS filtering, because all network traffic between Virtual Machine Monitor and Virtual Machines are passing through Security Supervisor.

![Figure 4: Centralized Security Supervisor between VMM and VMs](image)

A Virtual machine when affected by virus, threats or malware may sends requests to the Virtual Machine Monitor

- For receiving the more resources than it requires
- When it tries to access the resources of other virtual machines
• When it tries to get the memory of other virtual machines
• It may try to corrupt the data belongs to other virtual machines
• When it tries to disturb the normal functioning of other virtual machines.
• Sometimes it may try to disturb the function of Virtual machine monitor.

All the requests pass through the **Security Supervisor**. The security supervisor checks the details of virtual machines and its other attributes for validation. Thus it can distinguish malicious virtual machine from non-malicious one. If the virtual machine request is not in the normal form with regular required resources, then it does not satisfy the security check of the supervisor, then it will be treated as infected or malicious, and that request will not be allowed to fulfil. It will not get the permission to access the resources. If it is recognized as a valid request from the virtual machine, then it will be allowed to access the required resources. Before provisioning the required resources to the virtual machine, it is to be checked against its access control policy. Each and every virtual machine will have access control policies. If the virtual machine requesting for resources which is not permitted in its access control policy, then that request is not fulfilled (though that request is valid according to security supervisor). This method will provide security to virtualization environment by avoiding infected and malicious virtual machines resource requests, and allows only valid virtual machines to access the resources.

Security supervisor is designed in such a way that it should employ diverse kinds of security appliances, problem identification tasks, problem rectification means, eluding controlled users and accesses, process of walloping the originality of data mechanisms, approaches of recognizing and eluding intruders, malwares, viruses, functions that maintain the status of the virtual machines, their requesting patterns, behavioral patterns of virtual machines in different situations.

It may also contain some updates and patches (Patches are nothing but piece of codes, if the softwares or the programming codes are suffering from some problem or if there is a requirement of some additional code which makes the program to run normally, then to overcome such type of problems patches will be applied in the proper location of the software or code. Sufficient knowledge is required to apply the patches and also we should know the types of patches, which patch is for what problem, where to apply, when to
apply, location of the patch to apply, whether it is updated patch or not. If we are not in the position to decide which patch to apply for the present problem, instead of solving the problem, the applied patch may create new problems) needed by the virtual machines and softwares. It have a duty to have the elasticity of adding the up-to-date safety practices and additional solutions which may work very well to overcome the problems which may arise in the coming days. It should have the repetitive work of reshaping and testing its security checks, tools and solutions and keep updating at consistent interludes so that it should constantly hold newest, acceptable and active safety solutions. It may be given the flexibility that, some safety solutions can directly vaccinated to the virtual machines itself such as IDS/IPS, anti-virus kits etc.,

Some of the limitations of this approach are:

- Performance disputes may occur while realizing such type of safety solutions inside the Security supervisor. This is the main limitation of this approach, because all the requests by a number of virtual machines must pass through this Security supervisor for the purpose of security check. Security supervisor has to handle all those requests and also provision the resources based on their request and access control policy of each virtual machine, it has to check the status of each virtual machine when a request arrives for some resources. It has to check the access patterns, behavioral patterns of the virtual machines and also to check the access control policy of each virtual machine for each request it received. This becomes the bottleneck for the security supervisor.

- One more major issue is that, the response time of the virtual machine monitor will be increased because of the time required to perform different checks such as, authentication, authorization, validity checks, access control privileges check, status check, access patterns and behavioral patterns check for each request which are coming from the different virtual machine. Since our objective is to achieve security of the virtual machine environment, we need to tolerate for more response times of virtual machine monitor.

This is also simulated using the cloudsim tool. Here the limitations of the first approach is eliminated by adding Security supervisor which handles both infected requests as well as requests for more resources than they actually required. Here we make VMs to request for
more computing resources as well as we introduce some malwares into the VMs so that they send infected requests to VMM. This can be identified by Security supervisor effectively without giving chance to affect others.

4.3 Virtualized Environment with SS and IDS/IPS system (3rd Approach):

The third approach is to combine both first and second approaches, so that we have more levels of security checks for the virtual environment to avoid the security problems, which is shown in the figure given below:

In this approach, the burden of Security supervisor is reduced when compared to previous approach. Because external threats and malwares, virus problems to the virtual machines will be managed by the virtual machine themselves with the support of IDS/IPS system. The Security supervisor is only responsible for handling authorization and authentication, checking for access control privileges, status check of virtual machines, access patterns and behavioral checks of virtual machines, and check for valid or invalid requests coming from the different virtual machines. In this approach also there may be performance issues, because of Security supervisor functionality, though virtual machines have the responsibility of handling malwares, threats and viruses themselves, still it takes
much time to check the requests for validity and access control privileges, access patterns, behavioral patterns etc., Because of these types of different checks, the response time of the virtual machine monitor may be increased.

This approach is also simulated using cloudsim tool. Here we have two levels of security layers. At the first level IDS/IPS system is directly installed in the VMs directly, so that any threats or malwares entry into VMs are directly avoided at the initial level. And also it reduces the burden of Security supervisor which we encountered in 2nd approach. In addition the SS is able to handle the interactions between VMM and VMs effectively by making VMs to send only normal requests for the resources. If it finds the requests are not normal, then it may stop those requests to make VMM to not allocate the resources to such VM. Usually the abnormal requests are created in such a way that a code is written to make open the browsers frequently so that it consumes more processor time as well as memory so that other machines may suffer from lack of computing resources and sometimes they may stop processing due to lack or required computing resources.
4.4 Duplication Mechanism without Security system (4th Approach):

In the fourth approach, we have different technique to achieve the security of the virtual environment. Here we are going to use Duplication mechanism, by which even if one of the virtual machine infects due to external threats or for some other reason, if that virtual machine is not in the position to process, then the user is no need to worry about it, because we have several instances of the same virtual machine. We can make use of one of the instances to perform the same processing, without stopping the work, which is shown below:

![Diagram of Duplication mechanism without Security system]

Figure 6: Duplication mechanism without Security system

In this approach, after creating the required number of virtual machines, instead of using those virtual machines for the processing, we are going to apply one more technique called Duplication. Here we are going to create the instances of virtual machines, for example, 3 instances of Virtual Machine1 (VM1), 2 instances of Virtual Machine2 (VM2) and 3 instances of Virtual Machine3 (VM3). Now instead of using the Virtual Machines directly for processing, we are going to use their one of the instances for the same processing. Since we have created many instances for the same Virtual Machine, if one of the instances is infects or corrupts due to some reason, we have still
some more instances of the same Virtual Machine, we can continue the work without stopping by using one of the remaining instances. This makes the user can feel the fault tolerance capability of the virtual environment without any security system.

In this system, when the processing starts, instance 1 of virtual machine 1, virtual machine 2 and virtual machine 3 involve in processing of the respective virtual machines. In any case, if instance 1 of the virtual machine 1 fails to process due to some external threats, malwares or viruses problem, then the same work can be continued without stopping by using instance 2 of virtual machine 1. The same principle applies to all the virtual machines of the environment. If anyone instance of any virtual machine fails due to any reason, then the work of that instance can be continued without any problem by using one of the available instances of the same virtual machine. Since we create some number of instances for each virtual machine, the user will get the benefit of having the fault tolerant feature.

The leading problem of this approach is that, the system needs added memory for the creation of more number of instances for each virtual machine. But the main advantage which is considered as the main limitation of the previous approach is that, the response time of the Virtual machine monitor will be reduced. This is because, the virtual machine monitor will not employ any security checking mechanism, which consumes more time for performing various security checks. The requests of the virtual machines can be get provisioned with minimum time, so that the processing will be done faster by the instances of each virtual machine.

One more limitation of this approach is that, it does not contain any security mechanism to avoid the threats, malwares and viruses from inside or outside the environment. But this problem can be limited to some extent by having fault tolerant feature.

This approach is also simulated using cloudsim tool. In this approach we have one more component is introduced by which instances of VMs are created using the system call “Fork” of Java platform. Fork is used to create the instances of VMs and make them to work on behalf of VMs. In this approach also we make instances to send requests for more resources than they actually required and make instances to infect.
4.5 Duplication Mechanism with SS and IDS/IPS system (5th Approach):

In the fifth approach, we are combining the security features which are explained in first and second approach with the fourth approach, so that we can have both fault tolerance capability with required level of security for the virtual environment of the cloud services, which is shown below:

![Diagram of Duplication Mechanism with Security System]

Figure 7: Duplication Mechanism with Security System

In this approach, we have both Fault tolerance capability as well as security system for the virtual environment. This makes the user to have a feel that they have secured and fault tolerant processing environment. Here the main advantages are Security and fault tolerant features and the instances of the virtual machines will not be get affected by viruses, threats or malwares, because they equipped with intrusion detection and prevention systems, so that they can protect themselves from the inside or outside threats. But the major limitations are:

Firstly, it consumes more memory for the purpose of creating more number of instances for each virtual machine like previous approach. It requires almost two to three times more memory than the memory requirements of first three approaches.
Secondly, the response time of the Virtual machine monitor will be increased, this is because virtual machine monitor is employed with many security checking mechanisms such as authentication, authorization, validity checks, access patterns check, access control privileges and behavioral patterns check etc., to provide the security to the virtual machine environment.

In this system we can obtain one very important benefit. We can reduce the time taken by the virtual machines (Instances of virtual machines) for the purpose of processing. This can be done by allowing all the instances of the virtual machines to run concurrently to process, by dividing the task to be processed into different sub tasks. As we explained earlier in this approach, we have many instances for each virtual machine like virtual machine 1 has three instances, virtual machine 2 has two instances and virtual machine 3 has three instances. At any situation, only one instance of the virtual machine will be in processing state, if that instance fails due to some problems, then the same work can be continued with the one of the remaining instances of the virtual machine. Here only one instance will be in active state (processing) and remaining instances will be idle. That means we are wasting the memory for creating more instances and let them become idle most of the times. These instances can be used only when the failure of active instance occurs, otherwise the memory used to create these instances will be simply wasted. But here we have one more solution to these idle instances.

When the virtual machine receives its work to be done from the virtual machine monitor with all the required resources, instead of assigning that whole work to one of the instances of that virtual machine, it should divide the work in to sub modules, so that the number of sub modules should be equal to the number of instances that virtual machine is having. Then all instances should start processing at the same time and finish the processing with less time. This is how we can allow all the instances to process simultaneously by sharing the work among all the instances there by reducing the processing time and finishes the processing as early as possible when compared to previous method explained in this approach. Here the time required to processing will be reduced by 2-3 times by creating 2-3 instances for each virtual machine.

This approach is also simulated using cloudsim tool. We are using the Fork system call to create the virtual instances of the VMs in this approach like 5th approach and security.
system is also installed here to increase the level of security. Remaining parameters are same as 5th approach.

4.6 Secured Virtualized Environment (6th Approach):

In the sixth and last approach, we have a different mechanism for providing the security to the virtualized environment, which is shown below:

![Figure 8: Secured Virtualization Environment](image)

In this approach, we have four new components which plays very important role in deciding the virtual machines status, such as to check whether the virtual machines are working fine or not, whether their request is valid or not and whether they are infected or not. All the activities performed by each of the virtual machine is monitored and recorded in a component called Virtual Security Monitor (VSEM). This is essential to protect the virtual environment system from the malicious attacks while the communication takes place between Virtual machine monitor and virtual machines. In order to enhance security of the system, Virtual Security Monitor (VSEM) and Virtual Reliability Monitor (VREM) are introduced. These are present in each virtual machine. They interact with Hyper Security Monitor (HSEM) and Hyper Reliability Monitor (HREM) which are present in and accessed by Hypervisor (Virtual Machine Monitor). HSEM collects the information from VSEM and HREM to decide whether virtual machine is under attack or not.
VSEM present in every Virtual Machine, Monitor the behavior of Virtual Machine and collects the information about it from its interaction with Hypervisor, which helps to decide whether it is malicious or not. It monitors the transaction of respective Virtual Machine with Virtual Machine Monitor and lists all successful and failed transactions. At regular intervals this information is sent to HSEM. VREM maintains a threshold value for number of unsuccessful transactions, if any Virtual Machine tries to access the resources from the hypervisor and the transaction is failed, then the counter of VREM is incremented for each failed transaction. This counter value will be sent to HREM at regular intervals. If the counter crosses the threshold value, then the abnormal behavior of respective Virtual Machine is noted and considered it as victim. HREM decides which Virtual Machine is crossing the threshold. Then by using the information of HREM and VSEM, HSEM decides whether to satisfy the requests of Virtual Machines are not and whether to allow continue the processing or not.

This is the only approach which is not simulated using Cloudsim tool. Actually we tried to simulate this approach also, but we find difficulty in implementing the components called VSEM & VREM inside the Virtual machines and HSEM & HREM inside the VMM. There are some limitations in implementing these components inside the VMs and VMM. We tried it too, but VMs and VMM will not work according to requirements. They behave just like HSEM, HREM, VSEM and VREM not VMM or VMs.